



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/754,785	01/04/2001	Pierre-Alain Darlet	40101/06901	3238
30636 7590 08/09/2012 FAY KAPLUN & MARCIN, LLP 150 BROADWAY, SUITE 702 NEW YORK, NY 10038				
EXAMINER				
RUTTEN, JAMES D				
ART UNIT		PAPER NUMBER		
2197				
MAIL DATE		DELIVERY MODE		
08/09/2012		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte PIERRE-ALAIN DARLET

Appeal 2010-004717
Application 09/754,785
Technology Center 2100

Before LANCE LEONARD BARRY, ST. JOHN COURTENAY III, and
ANDREW J. DILLON, *Administrative Patent Judges*.

COURTENAY, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF CASE

Appellant appeals under 35 U.S.C. § 134 from a rejection of claims 1-15 and 40-60. Claims 16-39 have been cancelled. We have jurisdiction under 35 U.S.C. § 6(b).

We affirm.

Invention

The disclosed invention is directed to a method and system for “receiving a software module, the software module including references, [where] at least some of the references are backward references.” (Spec. 3.) The method includes “reordering the software module to remove at least some of the backward references.” (*Id.*).

Claim 1, reproduced below, is illustrative of the claimed subject matter:

1. A computer readable storage medium including a set of instructions executable by a processor, the set of instructions operable to:
 - receive a software module, the software module including references to locations within the software module, at least some of the references being backward references; and
 - reorder components of the software module into a predetermined order based on a type of the components* to remove at least some of the backward references,

wherein the components include at least one of a header, a section, and a table,

wherein the reordered software module includes the at least some of the backward references, and

wherein the at least some of the backward references in the reordered software module are stored in a memory to avoid a nonsequential reading of the reordered software module.
(disputed limitations emphasized).

PRIOR DECISION

Appeal No. 2007-0224 (Application No. 09/754,785) mailed May 23, 2007. (Examiner Affirmed (§§ 102, 103), New Grounds of Rejection under § 101 set forth by Board Panel under the authority of 37 C.F.R. § 41.50(b)).

REJECTIONS

1. Claims 1-15, 40, 41, and 43-60 stand rejected under 35 U.S.C. 103(a) as being unpatentable over John Levine, LINKERS AND LOADERS, chapter 6, June 1999, online accessed 08/15/2005, *available at* <http://www.iecc.com/linker/linker06.txt> (hereinafter “Levine”).

2. Claim 42 stands rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Levine and Breslau (U.S. Patent Number 6,185,733 B1, issued February 6, 2001).

GROUPING OF CLAIMS

Based on Appellant’s arguments in the Briefs, we decide the appeal on the basis of claims 1 and 42. *See* 37 C.F.R. § 41.37(c)(1)(vii).

FACTUAL FINDINGS

We adopt the Examiner’s factual findings as set forth in the Answer (Ans. 4, *et seq.*).

Appellant's Contentions

Appellant contends “that no ‘predetermined order’ is disclosed or suggested by Levine, and that the ordering disclosed by Levine is not ‘based on a type of the components,’ and, therefore, Levine does not disclose or suggest ‘reorder[ing] components of the software module into a predetermined order based on a type of the components,’ as recited in claim 1.” (App. Br. 5).

Independent claim 1

Issue: Under § 103, did the Examiner err in finding that the cited references, either alone or in combination, would have taught or suggested “*reorder[ing] components of the software module into a predetermined order based on a type of the components,*” within the meaning of claim 1 (emphasis added)?

ANALYSIS

At the outset, we observe that Appellant supports the aforementioned contentions by referring to an “exemplary embodiment of Figures 1-4 of the Specification” (App. Br. 4), and further contending that “[w]hile other *orders* may be possible in other exemplary embodiments, one of skill in the art will understand that *this order* is clearly predetermined and based on the functions of various components, as recited in claim 1.” (App. Br. 5, emphasis added).

The Examiner disagrees:

Appellant appears to contend that, “based on a type of the components,” must be interpreted as meaning “based on the functions of the various components,” (Br. 4-5 (emphasis

added) (citing Specification, pp. 5-17, Figs. 1-4).) Despite appellant's contrary assertion, claim 1 does not recite the reordering being, "based on the functions of various components." (See Br. 5; Claim 1.)

(Ans. 11).

We agree with the Examiner, and decline Appellant's invitation to read limitations from the Specification into the claims. "[A]lthough the specification often describes very specific embodiments of the invention, we have repeatedly warned against confining the claims to those embodiments." *Phillips v. AWH Corp.*, 415 F.3d 1303, 1323 (Fed. Cir. 2005) (en banc) (citations omitted). A basic canon of claim construction is that one may not read a limitation into a claim from the written description. *Renishaw PLC v. Marposs Societa' per Azioni*, 158 F.3d 1243, 1248 (Fed. Cir. 1998). Thus, Appellant's argument ("this order is clearly predetermined and based on the functions of various components," App. Br. 5 (emphasis added)) is unconvincing because it is not commensurate with the scope of the claim.

In broadly but reasonably construing the scope of the disputed "predetermined order," even when we consider (as *context*) the portion of the Specification Appellant points to for support (App. Br. 2; Spec. p. 5, ll. 25-29), we find only a non-limiting exemplary description of how the claimed *predetermined order* is generated based on a type of the components:

In step 104, in accordance with the present invention, the software module is processed into an alternative format, eliminating many of the backward references in the file, and *placing the components of the software module in a predetermined order*. In the *example embodiment*, this may be accomplished as a separate step using the procedure describe[d]

below. However, *skilled practitioners of the pertinent art will recognize that other procedures of generating a software module in the alternative format are possible.*

(Spec. 5, ll. 25-29) (emphasis added).

In the Reply Brief, Appellant acknowledges that Levine teaches reordering, “but without reference to the type of object files being reordered.” (Reply Br. 4). In particular, Appellant contends that “*Levine* does not provide any support for the notion that these object files cover *multiple types* of such files.” (*Id.*) (emphasis added).

However, we find Levine’s files are at least of type “object file.” (Levine 5). We conclude that the broad language of claim 1 does not positively recite that *multiple types* are required: “reorder[ing] components of the software module into a predetermined order based on a type of the components . . .” (claim 1)(emphasis added). Thus, we conclude that the language of claim 1 broadly encompasses *selectively* reordering *only* a plurality of components of a (*single*) particular type (e.g., object file type) into a predetermined order *according to any method*.

Because the ordering in Levine (by the “lorder” program) is disclosed as *only* being applied to the input set of *object file* type, we find Levine’s (re)ordering is “based on a type of the components,” within the meaning of claim 1. (Levine 5). Because Levine’s sorted list of object files (“tsort” topological sort on the output of “lorder”) has “each symbol [] defined after all the references to it,” (i.e., all backward references are removed) (*id.*), we find Levine teaches “remov[ing] at least some of the backward references,” within the meaning of claim 1.

On this record, Appellant has not persuaded us that the Examiner's construction of the disputed claim term "predetermined order" is overly broad, unreasonable, or inconsistent with the Specification. For essentially the same reasons articulated by the Examiner (Ans. 10-13), and for the reasons discussed above, we are of the view that the disputed limitation of "*reorder[ing] components of the software module into a predetermined order based on a type of the components*" (claim 1) (emphasis added) is at least suggested by the cited combination of Levine and Breslau.

For these reasons, we are not persuaded of error regarding the Examiner's underlying factual findings and ultimate legal conclusion of obviousness. Accordingly, we sustain the Examiner's obviousness rejection of independent claim 1.

Independent claims 9 and 55

Regarding claims 9 and 55, Appellant merely recites the language of these claims and contends that these claims are allowable for the same reasons previously advanced for claim 1. (App. Br. 5-6). However, "[a] statement which merely points out what a claim recites will not be considered an argument for separate patentability of the claim." 37 C.F.R. § 41.37(c)(1)(vii). Therefore, we sustain the Examiner's obviousness rejection of independent claims 9 and 55 for the same reasons discussed above regarding claim 1.

Dependent claim 42

Regarding claim 42, Appellant “submits that Breslau does not cure the deficiencies of Levine discussed above with reference to claim 1.” (App. Br. 6-7). However, we are not persuaded of any deficiencies regarding Levine for the reasons discussed above regarding claim 1. Therefore, we sustain the Examiner’s obviousness rejection of claim 42 for the same reasons discussed above regarding claim 1.

Remaining Dependent claims

Regarding the remaining dependent claims, Appellant presents no separate arguments. Arguments not made are considered waived.¹ Therefore, the remaining dependent claims fall for the same reasons discussed above regarding the respective independent claims.

DECISION

We affirm the Examiner’s § 103 rejections of claims 1-15, and 40-60.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a). *See* 37 C.F.R. § 41.50(f).

ORDER

AFFIRMED

¹ *See Hyatt v. Dudas*, 551 F.3d 1307, 1314 (Fed. Cir. 2008) (“When the appellant fails to contest a ground of rejection to the Board, section 1.192(c)(7) [(now section 41.37(c)(1)(vii))] imposes no burden on the Board to consider the merits of that ground of rejection [T]he Board may treat any argument with respect to that ground of rejection as waived.”).

llw